AD-A107 620 MISSOURI UNIV-COLUMBIA DEPT OF STATISTICS F/G 12/1
AN ISOTONIC REGRESSION ALGORITHM.(U)
JUL 81 R L DYKSTRA N00014-78-C-0655
UNCLASSIFIED TR-106 NL

1 of 1
AD
A107620

END
DATE
FILMED
1 82
DTIC

LEVEL II

(14)

AD A107620

# University of Missouri-Columbia

**An Isotonic Regression Algorithm**

DTIC ELECTE NOV 19 1981 S D E

Richard L. Dykstra
University of Missouri-Columbia

Technical Report No. 106
Department of Statistics July 1981

## Mathematical

## Sciences

DTIC FILE COPY



81 11 18 066

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

| REPORT DOCUMENTATION PAGE | | READ INSTRUCTIONS BEFORE COMPLETING FORM |
|---|---|---|
| 1. REPORT NUMBER<br>106 | 2. GOVT ACCESSION NO.<br>AD-A107620 | 3. RECIPIENT'S CATALOG NUMBER |
| 4. TITLE (and Subtitle)<br>An Isotonic Regression Algorithm | | 5. TYPE OF REPORT & PERIOD COVERED<br>Technical Report |
| | | 6. PERFORMING ORG. REPORT NUMBER |
| 7. AUTHOR(s)<br>Richard L. Dykstra | | 8. CONTRACT OR GRANT NUMBER(s)<br>N00014-78-C-0655 |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS<br>Department of Statistics<br>University of Missouri<br>Columbia, MO. 65211 | | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS |
| 11. CONTROLLING OFFICE NAME AND ADDRESS<br>Office of Naval Research<br>Department of the Navy<br>Arlington, VA. | | 12. REPORT DATE<br>June 1981 |
| | | 13. NUMBER OF PAGES<br>14 |
| 14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) | | 15. SECURITY CLASS. (of this report)<br>Unclassified |
| | | 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE |

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release: Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Isotonic regression, quasi-order, minimal violator, maximal violatee, least squares, pooling algorithm.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A "pooling sets" type of algorithm is developed and shown to be valid for computing an isotonic regression function for a general quasi-order. The method is direct and intuitive. The algorithm works best when the quasi-order is complex and the objective function is nearly isotonic. An example is worked out in detail.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

# AN ISOTONIC REGRESSION ALGORITHM

Richard L. Dykstra
University of Missouri-Columbia



This work was supported in part by ONR Contract N00014-78-C-0655.

## ABSTRACT

A "pooling sets" type of algorithm is developed and shown to be valid for computing an isotonic regression function for a general quasi-order. The method is direct and intuitive. The algorithm works best when the quasi-order is complex and the objective function is nearly isotonic. An example is worked out in detail.

Key Words: Isotonic regression, quasi-order, minimal violator, maximal violatee, least squares, pooling algorithm.

AMS Classification number 90C99

## 1. Introduction.

Consider $X$ to be a finite set which we denote as $X = \{x_1, x_2, \ldots, x_m\}$. Let $\lesssim$ be a binary relation on $X$ which is a quasi-order, i.e. $\lesssim$ is reflexive and transitive, (see Barlow, Bartholomew, Bremner, and Brunk (1972) p.24). We say a real-valued function $f(\cdot)$ defined on $X$ is isotonic with respect to "$\lesssim$" if $x \lesssim y$ implies $f(x) \leq f(y)$. A problem that arises in many different contexts is the following: For a given positive function $w(x)$, called the weight function, and an arbitrary function $g(x)$, find that function $g^*(x)$ on $X$ which is isotonic with respect to a particular quasi-order "$\lesssim$" and minimizes

$$\sum_{i=1}^{m} (g(x_i) - h(x_i))^2 w(x_i)$$

over all isotonic functions $h(x)$. This is referred to as the isotonic regression problem. Much of the importance of this problem hinges on the fact that is solves many seemingly very different optimization problems (see Barlow et.al. (1972) and Barlow and Brunk (1972)).

This is of course a quadratic programming problem with special types of linear constraints. In certain cases, the solution is well known and easily computed.

i) Simple ordering case. When "$\lesssim$" is also a simple ordering (reflexive, transitive, anti-symmetric, and each pair of elements is comparable), the problem is greatly simplified. One of the more common situations is when $X$ consists of real numbers and "$\lesssim$" is the usual ordering of the real numbers. In this case, a function is isotonic iff it is nondecreasing. In this situation,

$g^*(x)$ can be effected in the following way: Reduce the problem to fewer dimensions by pooling adjacent points where reversals occur. That is, choose $x_i, x_j$ such that $x_i \lesssim x_j$, $g(x_i) > g(x_j)$, and there does not exist $y$ such that $x_i \lesssim y \lesssim x_j$ (if no such pair exists, then $g(x) = g^*(x)$). Now consider the same problem where $x_i$ and $x_j$ are considered as one point with weight $w(x_i) + w(x_j)$, functional value

$\tilde{g}(\{x_i, x_j\}) = [g(x_i)w(x_i) + g(x_j)w(x_j)][w(x_i) + w(x_j)]^{-1}$,

and the natural ordering $\{x_i, x_j\} \lesssim x_k$ if $x_j \lesssim x_k$ and $x_k \lesssim \{x_i, x_j\}$ if $x_k \lesssim x_i$. This reduction process is continued until a function $\tilde{g}$ is obtained with no reversals. Then $g^*(x) = \tilde{g}(\hat{x})$ if $x$ is one of the points making up the pool $\hat{x}$. This is the "Pool Adjacent Violators Algorithm" first discussed by Ayer et.al. (1955).

ii) Rooted trees. A more general class of partial orderings that is important in some problems involves the concept of rooted trees. In a rooted tree, except for a single element, the root, which has no predecessor, each element has exactly one immediate predecessor. (We say $x_j$ is an immediate predecessor of $x_i$ if $x_j \lesssim x_i$ and there does not exist an element $y$ such that $x_j \lesssim y \lesssim x_i$).

Thompson (1962) has shown that the previous scheme used for the simple ordering case works providing the poolings are taken in the correct order. To be more precise, if there exists an immediate predecessor to $x_i$, say $x_j$ such that $g(x_j) > g(x_i)$, call the point $x_i$ a violator (of $x_j$) and call $x_j$ a violatee (of $x_i$). A violator $x_i$ is a minimum violator if there does not exist a violator $x_k$ such that $g(x_k) < g(x_i)$. Thompson showed that if one pools a minimum violator to its violatee (there exists only one in the

rooted tree case), redefines the partial order in the natural way, and continues as in the case of simple ordering, that one must eventually arrive at $g^*(x)$. This is often called the "Minimal Violator Algorithm".

iii) <u>Partial ordering case</u>. For general partial orders, the solutions become less tractable. A closed form expression for $g^*$ described in Brunk (1955) involves the use of upper and lower sets.

a) $A \subset X$ is an upper set if $x_i \varepsilon A$ and $x_i \lesssim x_j$ implies $x_j \varepsilon A$.

b) $A \subset X$ is a lower set if $x_i \varepsilon A$ and $x_j \lesssim x_i$ implies $x_j \varepsilon A$.

If we let $\mathcal{U} = \{A;\ A \text{ is an upper set}\}$, $\mathcal{L} = \{A;\ A \text{ is a lower set}\}$, and for any $A \subset X$

$$Av(A) = \frac{\sum_{t \varepsilon A} g(t)w(t)}{\sum_{t \varepsilon A} w(t)},$$

then

$$g^*(x) = \max_{\substack{A\varepsilon\mathcal{U} \\ x\varepsilon A\cap B}} \ \min_{B\varepsilon\mathcal{L}} \ Av(A \cap B). \qquad (1.1)$$

Based on this expression for $g^*(x)$, there exists a simple algorithm (Minimum Lower Sets Algorithm) stated in Brunk, Ewing, Utz (1957) for computing $g^*$. To use this algorithm, first find $L_1 \varepsilon \mathcal{L}$ such that $Av(L_1)$ is minimal. Then for $x \varepsilon L_1$, $g^*(x) = Av(L_1)$. Next find $L_2 \varepsilon \mathcal{L}$ such that $Av(L_2 - L_1)$ is minimal. Then for $x \varepsilon L_2 - L_1$, $g^*(x) = Av(L_2 - L_1)$, etc. This is repeated until $\cup L_i = X$ and $g^*$ is completely determined.

This algorithm is straightforward and direct, but does have some shortcomings.

1) The class of all lower sets may be difficult to determine

and enumerate and/or excessively large. For example, in the natural partial ordering for the grid $X = \{(i,j);\ i = 1,\ldots,a,\ j = 1,\ldots,b\}$ defined by

$$(i,j) \lesssim (k,\ell) \text{ iff } i \leq k \ \underline{\text{and}}\ j \leq \ell,$$

it can be shown by combinatorial techniques that there are $\binom{a+b}{a}$ lower sets. This number becomes very large as a and b become large.

2) Computing the many averages Av(L) for all lower sets may involve a large amount of effort.

3) If g is nearly isotonic (very few reversals), as much work (in fact usually more because the $L_1$, $L_2$, etc. are small and hence more minimums must be compared) is required as if g is far from isotonic.

*For these reasons*, a scheme similar to the pooling algorithm of Thompson for the rooted tree case would be desirable. Since an arbitrary quasi-ordering allows a great deal more diversity than the rooted tree situation, it is not surprising that the general quasi-order algorithm is more complicated than for the rooted tree case. As in Thompson's algorithm, poolings are again formed with the idea of reducing the dimension of the problem. However, in this case, we must also form temporary or "working" pools which may be dissolved later in the algorithm.

After the pooling, our new problem is defined in the natural way. That is, if the points x and y are pooled, then $\{x,y\}$ is treated as one point with weight $w(x) + w(y)$ and functional valve $Av(\{x,y\})$. The ordering is preserved naturally; i.e. $\{x,y\} \lesssim z$ if

$x \precsim z$ or $y \precsim z$ and $z \precsim \{x,y\}$ if $z \precsim x$ or $z \precsim y$.
We remark that poolings are to be made only between violators and their immediate predecessors.

## 2. Justification of Algorithm.

Since the algorithm works by pooling points together, we need some criterion which will justify using the pooling mechanism. Lemma 2.1 is basic to the algorithm. Lemma 2.1 is similar to Theorem 2.5 of Barlow et.al. (1972) except we need not assume that $x_1$ and $x_2$ are "poolable".

Lemma 2.1. If $g^*$ denotes the solution to an isotonic regression problem and $\hat{g}$ denotes the solution to the isotonic regression problem where $x_1$ and $x_2$ are pooled to form y (see section 1 for details of the pooling), then providing $g^*(x_1) = g^*(x_2)$, $\hat{g}$ and $g^*$ agree in the sense that $g^*(x) = \hat{g}(x)$, $x \neq x_1, x_2$ and $g^*(x_1) = g^*(x_2) = \hat{g}(y)$.

Proof. Suppose $h$ is a function defined on $X$ such that $h(x_1) = h(x_2) = c$ and $h'$ is a function such that $h'(x) = h(x)$, $x \neq y$, and $h'(y) = c$. Note first that $h$ is isotonic on $X$ iff $h'$ is isotonic for the pooled problem. Then the sum of terms in

$$\sum_{i=1}^{m} (g(x_i) - h(x_i))^2 w(x_i) \tag{2.1}$$

which involve $x_1$ and $x_2$ are

$$(g(x_1)-c)^2 w(x_1) + (g(x_2)-c)^2 w(x_2)$$

$$= (Av(\{x_1,x_2\})-c)^2 (w(x_1) + w(x_2)) + c(x_1,x_2)$$

where $\Phi$ does not depend upon c. Since the value of c that minimizes the left side also minimizes the right side, the assumption that $g^*(x_1) = g^*(x_2)$ implies the desired result.

Thus if we have points $x_1$ and $x_2$ such that $x_1 \lesssim x_2$ and $x_2 \lesssim x_1$, we know that we may pool $x_1$ and $x_2$ and consider the pooled problem.

An algorithm due to Von Eeden (see Barlow et.al (1972), page 90) is of interest to us. In this scheme, order restrictions are numbered sequentially, say $0_1, 0_2, \ldots, 0_p$. Let $g_k^*$ denote the solution to the smaller problem consisting of only the first k constraints. Then consider the $k+1^{st}$ constraint, say $h(x_i) \lesssim h(x_j)$. If $g_k^*(x_i) \leq g_k^*(x_j)$, $g_{k+1}^* = g_k^*$. Otherwise, $g_{k+1}^*(x_i) = g_{k+1}^*(x_j)$, which effectively reduces the problem of finding $g_{k+1}^*$ to one involving only k constraints. By induction, one can obtain $g^*$ by using the above result. The problem lies in the inefficiency of the method, since any pooling may be later broken up. With the observation that the minimal violator always stays pooled, we may greatly improve upon this scheme for the isotone regression problem.

If $A \subset X$, $g(x)|_A$ will denote g restricted to A. We can also consider our isotonic regression problem scaled down to A. That is, for $x_i$, $x_j \varepsilon A$, $x_i \lesssim x_j$ (in A) iff $x_i \lesssim x_j$ (in X). We will denote the solution of our restricted problem by $g_A^*$. Of course $g^*|_A$ is the overall solution restricted to A. We assume that $\lesssim$ is a partial order in the following three lemmas.

<u>Lemma 2.2</u>. If L is a lower set, then $g_L^* \geq g^*|_L$.

<u>Proof</u>. Note first that the upper and lower sets of our restricted problems are respectively

$\mathcal{U}_L = \{A \cap L;$ A is an upper set$\}$, and

$\mathcal{L}_L = \{B \cap L;$ B is a lower set$\}$.

Thus, from (1.1),

$$g_L^*(x) = \max_{\substack{A'\in\mathcal{U}_L \\ x\in A'\cap B'}} \min_{B'\in\mathcal{L}_L} Av(A' \cap B')$$

$$= \max_{\substack{A\in\mathcal{U} \\ x\in A\cap B}} \min_{B\in\mathcal{L}} Av(A \cap B \cap L)$$

$$\geq \max_{\substack{A\in\mathcal{U} \\ x\in A\cap B}} \min_{B\in\mathcal{L}} Av(A \cap B)$$

$$= g^*(x)|_L$$

since $B\in\mathcal{L} \Rightarrow B \cap L\in\mathcal{L}$ and hence $\mathcal{L}$ contains at least as many sets as $\mathcal{L}_L$

A lemma somewhat similar to lemma 2.2 which we shall need is the following one.

Lemma 2.3. If $A^* \cap B^* \subset C = A_1 \cap B_1$ where $A^*$ and $A_1$ are upper sets, $B^*$ and $B_1$ are lower sets, and for all $x\in A^* \cap B^*$, $g^*(x) = Av(A^* \cap B^*)$, then $g_C^*(x) = g^*(x) = Av(A^* \cap B^*)$ for $x \in A^* \cap B^*$.

Proof. For $x\in A^* \cap B^*$ we can write

$$g_C^*(x) = \max_{\substack{A\in\mathcal{U} \\ x\in A\cap B}} \min_{B\in\mathcal{L}} Av(A \cap B \cap A_1 \cap B_1)$$

$$\geq \max_{\substack{A\in\mathcal{U} \\ x\in A\cap B}} \min_{B\in\mathcal{L}} Av(A \cap A_1 \cap B)$$

$$= g^*(x), \quad \text{since } A^* = A^* \cap A_1.$$

Similarly,

$$g_C^*(x) = \min_{\substack{B\in L \\ x\in A\cap B}} \max_{A\in U} Av(A \cap B \cap A_1 \cap B_1)$$

$$\leq \min_{\substack{B\in L \\ x\in A\cap B}} \max_{A\in U} Av(A \cap B \cap B_1)$$

$$= g^*(x), \quad \text{since } B^* = B^* \cap B_1.$$

from which the conclusion follows.

Note that this lemma implies that if for any set B, $A = g^{*-1}(B)$, then $g^*(x)|_A = g_A^*(x)$. We will also need lemma 2.4.

<u>Lemma 2.4</u>. If $m_1$ is a minimal violator in $X$, $g^*(m_1) = g^*(v)$ for some immediate predecessor v of $m_1$ (i.e., $m_1$ can be pooled with one of its immediate predecessors.

<u>Proof:</u> Assume $g^*(m_1) > g^*(v)$ for all immediate predecessors v of $m_1$. Thus if we omit all restraints of the form $h(x_j) \leq h(m_1)$, the reduced problem still admits the solution $g^*$. Let

$$B = \{x;\ g(x) < g(m_1)\} \cup \{m_1\} .$$

Since $g(x) \geq g(m_1)$ for all $x \in \overline{B}$ , we must have $g_{\overline{B}}^*(x) \geq g(m_1)$ (see Brunk (1965)). Thus if we define

$$\hat{g}(x) = \begin{cases} g^*_{\overline{B}}(x), & x\in\overline{B} \\ g(x), & x\in B, \end{cases}$$

$\hat{g}(x)$ is isotonic for the reduced problem. To see this, note that

for the original problem, there can be no violators in B save $m_1$ since $m_1$ is a minimal violator, and hence, for the reduced problem, there are no violators in B. No point in B can be a violatee, since the corresponding violator would have to be in $\bar{B}$ and $g^*_{\bar{B}} \geq g(m_1) \geq g(z)$, $z \varepsilon B$. Thus $\hat{g}$ satisfies all restrictions involving at least one element of B and, from its definition, clearly satisfies all restrictions involving elememts both in $\bar{B}$. However, by Van Eeden's algorithm (see first part of this section), $\hat{g} = g^*$, which is a contradiction, since a violatee v of $m_1$ belongs to $\bar{B}$ and thus $g^*_{\bar{B}}(v) = g^*(v) < g^*(m_1) = g(m_1)$.

Thus the minimal violator must be permanently pooled with at least one of its immediate predecessors. The following theorem answers the question of "Which one?" and serves as the basis of our algorithm for computing g*. For any point x, we define the particular lower set $L(m_1)$ as being the complement of $\{y;\ x \precsim y\}$.

<u>Theorem 2.1</u>. If $m_1$ is a minimal violator and $v_1$ the maximal immediate predecessor of $m_1$ with respect to $g^*_{L(m_1)}$ (i.e. $g^*_{L(m_1)}(v_1) \geq g^*_{L(m_1)}(v)$ for all other immediate predecessors v of $m_1$), then $g^*(m_1) = g^*(v)$.

<u>Proof.</u> Assume $g^*(m_1) > g^*(v_1)$. From lemma 2.4, there must exist some immediate predecessor of $m_1$, say v, such that

$$g^*(m_1) = g^*(v). \tag{2.2}$$

Noting that $\{x; g^*(x)=g^*(v_1)\} \subset L(m_1)$, it follows from Lemma 2.3, that

$$g^*(v_1) = g^*_{L(m_1)}(v_1) \tag{2.3}$$

By definition,

$$g^*_{L(m_1)}(v_1) \geq g^*_{L(m_1)}(v) \ . \tag{2.4}$$

Finally, Lemma 2.2 implies

$$g^*_{L(m_1)}(v) \geq g^*(v) \ . \tag{2.5}$$

Combining (2.3), (2.4), (2.5) and (2.2), we have

$$g^*(v_1) = g^*_{L(m_1)}(v_1) \geq g^*_{L(m_1)}(v) \geq g^*(v) = g^*(m_1),$$

which is a contradiction.

## 3. The Algorithm.

Lemma 2.1 and Theorem 2.1 suggest a pooling algorithm for obtaining the solution to an isotane regression problem with a general quasi-ordering. That is first reduce the quasi-ordering problem to a partial ordering problem by sequentially pooling points x and y where $x \lesssim y$ and $y \lesssim x$ and redefining the ordering until all such pairs are coalesced.

If one can solve the partial ordering problem in n dimensions and is confronted with such a problem in n + 1 dimensions, he has only to locate a minimal violator $m_1$, solve the restricted problem over $L(m_1)$ and pool $m_1$ with the resulting maximal immediate predecessor (with respect to $g^*_{L(m_1)}$). This reduces the problem to n dimensions which can be handled by assumption. Actually the scheme becomes much more tractable when one realizes that one does not have to begin all over again for all lower points after a pooling is made. Lemma 2.3 and Lemma 3.1 are particularly useful in this regard.

Lemma 3.1. If A and B are disjoint and if

$$\hat{g}(x) = \begin{cases} g_A^*(x) , & x \varepsilon A \\ g_B^*(x) , & x \varepsilon B \end{cases} ,$$

then $g_{A\cup B}^*(x) = \hat{g}(x)$ if $\hat{g}$ is isotonic over the set $A \cup B$.

Proof. The proof is obvious by writing the sum to be minimized as the sum over A plus the sum over B.

We note from Lemma 2.3 that if v is the maximal immediate predecessor of m under $g_{L(m)}^*$, and if

$$A = \{y\varepsilon L(m): g_{L(m)}^*(y) \neq g_{L(m)}^*(v)\} ,$$

then $g_{L(m)}^*(x) = g_A^*(x)$ for all $x\varepsilon A$. By using this fact with lemma 3.1, it is often easy to construct $g_{L(m')}^*$ for the next minimal violator $m'$.

Of course one can also describe the algorithm by first finding maximal violatees, and then pooling these to their (restricted) minimal violators. For large sets, a combination of the two approaches is sometimes useful.

## 4. An Example.

Consider the sixteen points in the plane $X = \{(i, j); i,j = 1, 2, 3, 4\}$ with the usual planar ordering $(i, j) \lesssim (h, k)$ iff $i \leq h$ and $j \leq k$. Assume all weights are equal to one and that

g(i,j)

| j \ i | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 4 | .1 | 2 | 7 | 6 |
| 3 | .1 | 6 | 5 | 6 |
| 2 | 5.2 | 0 | 5.2 | 5.5 |
| 1 | 1 | 5 | 3 | 4 |

.

Since the minimal violator is (2,2) and

| | | $g^*_{L(2,2)}$ | | | |
|---|---|---|---|---|---|
| | 4 | 1.8 | | | |
| | 3 | 1.8 | | | |
| j | 2 | 1.8 | | | |
| | 1 | 1 | 4 | 4 | 4 |
| | | 1 | 2 | 3 | 4 |
| | | | i | | |

,

we know that (2,2) and (2,1) may be pooled to yield

| | | | | | |
|---|---|---|---|---|---|
| | 4 | 1.8 | 2 | 7 | 6 |
| j | 3 | 1.8 | 6 | 5 | 6 |
| | 2 | 1.8 | 2.5 | 5.2 | 5.5 |
| | 1 | 1 | | 3 | 4 |
| | | 1 | 2 | 3 | 4 |
| | | | i | | |

.

Now (2,4) is the new minimal violator and it is easily shown that

| | | $g^*_{L(2,4)}$ | | | |
|---|---|---|---|---|---|
| | 4 | 1.8 | | | |
| | 3 | 1.8 | 5.5 | 5.5 | 6 |
| j | 2 | 1.8 | 2.5 | 5.2 | 5.5 |
| | 1 | 1 | | 3 | 4 |
| | | 1 | 2 | 3 | 4 |
| | | | i | | |

.

Thus we pool (2,4) with (2,3) to obtain

| $j$ | | | | | |
|---|---|---|---|---|---|
| | 4 | 1.8 | 4 | 7 | 6 |
| | 3 | 1.8 | | 5 | 6 |
| | 2 | 1.8 | 2.5 | 5.2 | 5.5 |
| | 1 | 1 | | 3 | 4 |
| | | 1 | 2 | 3 | 4 |
| | | | $i$ | | |

(4.1)

Now, (3,3) is the new minimal violator with $g^*_{L(3,3)}$ as given in expression (4.1). Pooling (3,3) with (3,2), we then have (4,4) as the minimal (only) violator. Pooling (4,4) with (3,4) we have as the final solution

$g^*_{L(i,j)}$

| $j$ | | | | | |
|---|---|---|---|---|---|
| | 4 | 1.8 | 4 | 6.5 | 6.5 |
| | 3 | 1.8 | 4 | 5.1 | 6 |
| | 2 | 1.8 | 2.5 | 5.1 | 5.5 |
| | 1 | 1 | 2.5 | 3 | 4 |
| | | 1 | 2 | 3 | 4 |
| | | | $i$ | | |

## REFERENCES

(1) Ayer, M., Brunk, H. D., Ewing, G. M., Reid, W. T. and Silverman, E. (1955). An empirical distribution function for sampling with incomplete information. _Ann. Math. Statist._ 26, 641-647.

(2) Barlow, R. E., Bartholomew, D. J., Bremner, J. M., and Brunk, H. D. (1972). _Statistical Inference under Order Restrictions_. Wiley:New York.

(3) Barlow, R. E. and Brunk, H. D. (1972). The isotonic regression problem and its dual. _J. Amer. Statist. Assoc._ 67, 140-147.

(4) Brunk, H. D. (1965). Conditional expectation given a $\sigma$-lattice and applications. _Ann. Math. Statist._ 36, 1339-1350.

(5) Brunk, H. D. (1955). Maximum likelihood estimates of monotone parameters. _Ann. Math. Statist._ 26, 607-616.

(6) Brunk, H. D., Ewing, G. M. and Utz, W. R. (1957). Minimizing integrals in certain classes of monotone functions. _Pacif. J. Math._ 7, 833-847.

(7) Thompson, W. A., Jr. (1962). The problem of negative estimates of variance components. _Ann. Math. Statist._ 33, 273-289.

DAT
ILM